Preprint version of the paper published in Expert Systems with Applications, doi:10.1016/j.eswa.2011.01.150

Understanding the Role of Conceptual Relations in Word Sense Disambiguation

David Fernandez-Amoros

Dpto. de Lenguajes y Sistemas Informaticos
david@lsi.uned.es
Ruben Heradio
Dpto. de Ingenieria de Software y Sistemas Informaticos
rheradio@issi.uned.es
Universidad Nacional de Educacion a Distancia
Madrid, Spain

Abstract

In this article, we concentrate in conceptual relations as a source of information for Word Sense Disambiguation (WSD) systems. We start with a review the most relevant research in the field, then we implement our own algorithm. As a starting point we have chosen the conceptual density algorithm of Agirre and Rigau. We generalize the original algorithm, parameterizing many aspects. This new algorithm obtains a relative improvement of 24% in terms of precision and recall. We also offer comparative evaluation of our system with respect to the participants in the SENSEVAL-2 disambiguation competition.

We conclude that conceptual relations provide a source of information that is insufficient by itself to achieve good disambiguation results, but can, however, be a very accurate heuristic in a combined system.

1 Introduction

Word Sense Disambiguation (WSD) is the problem of determining, for a word in context, the specific meaning of the word in a dictionary, or, more generally, a sense inventory. The classical example would the word bank in sentences like *I asked for a loan in the bank* or *I sat in the bank of the river*. There are several ways to evaluate the performance of a WSD disambiguation system. Precision is informally the accuracy of the system over the words it has been able to disambiguate. Recall is the measure of the performance of the system overall. With the appropriate definition, recall = precision · coverage. Coverage is the ratio of disambiguated words (correctly or not) over the total number of words.

These measures are often computed comparing the results of a system with a gold standard that has been disambiguated by hand. WordNet [Miller, 1995] is a lexical knowledge base that is also a popular sense inventory. The SemCor collection [Francis and Kucera, 1967] is a subset of the Brown Corpus tagged by hand.

Much work in WSD has used WordNet and SemCor as sense inventory and gold standard respectively, for instance [Agirre and Martinez, 2001, Agirre and Rigau, 1995, Agirre and Rigau, 1996, Voorhees, 1993, Sussna, 1993, Chodorow et al., 2000, Cucchiarelli et al., 2000, Dini et al., 1998, Dorr and Jones, 1996, Fellbaum et al., 1997, Haynes, 2001, Krovetz, 1998, Kwong, 2001, Lin, 1997, Mihalcea and Moldovan, 2000a, Mihalcea

Preprint version of the paper published in Expert Systems with Applications, doi:10.1016/j.eswa.2011.01.150

and Moldovan, 1999, Mihalcea and Moldovan, 2000b, Moon, 2000, Ng and Zelle, 1997, Ng, 1997, Ng and Lee, 1996, Montoyo and Suárez, 2001, Stevenson and Wilks, 1999, Banerjee and Pedersen, 2002, Patwardhan et al., 2003] to name but a few.

The SENSEVAL competition is an open sense evaluation conference that takes place periodically, which consists in several disambiguation tasks for various languages. Some of these tasks are provided with training data and test and evaluation collections. This competition has also contributed to confirm the position of WordNet as a standard sense inventory.

A competitive level in WSD, as proven by the participants in the first SENSEVAL [Kilgarriff and Rosenzweig, 2000] can only be attained combining knowledge sources of several kinds : Coocurrence information, syntactic information, collocations, additional information from dictionaries such as domain labels, selectional restrictions and all sorts of heuristics, see for example [Ng and Lee, 1996, Rigau et al., 1997, Wilks and Stevenson, 1998, Stevenson and Wilks, 1999]. A problem with such hybrid systems is that it is difficult to discern which is the discriminating power of each of the different kinds of knowledge about the context to disambiguate. It is our opinion that a separate, detailed study of each knowledge source is a necessary step in order to understand the challenges of WSD.

In this article, we concentrate in conceptual relations as source of information for WSD systems. The basic hypothesis is that the correct senses for the words in a text written in a natural language have closer relations (in a semantic network) that incorrect sense combinations. For instance, in *Spring is my favorite season*, the *springtime* sense of *spring* has a hyponymy relation with the *season of the year* sense of *season*, while as any other combination of senses (for instance *spring* as fountain and *season* as sports season) has weaker semantic relations.

It is our intention to undertake an in-depth study (through an exhaustive evaluation) of the role that conceptual relations can play with respect to accurate WSD. As a starting point, we choose one the most promising WSD algorithms based on a conceptual density measure [Agirre and Rigau, 1996]. As in that research, we have used the semantic network of WordNet, as a lexical database that provides senses for words and semantic relations between them. WordNet-1.7 includes 192460 senses of words for English and there are large-scale versions for many other languages, mainly EuroWordNet [Vossen, 1998].

We start by generalizing the algorithm, parameterizing many aspects of the original one, including the density formula itself. The new strategies that we have incorporated to the algorithm include all the possibilities we could think of. Next we carry out an exhaustive evaluation, running the algorithm in a sample of all the possible different configurations against all the nouns in the SemCor collection, the biggest hand-annotated collection that we are aware of. The original algorithm by Agirre and Rigau had not been tested against the whole collection.

Since the algorithm that we present here depends on several parameters, there is considerable variability of (precision, recall) pairs. We have opted for optimizing recall, unless otherwise stated, to find out what sort of results could be obtained if this source of information was used on its own. We have also been interested in researching the conditions under which the algorithm would achieve the best precision results so as to be used as a high-precision heuristic in a more complex system.

This first evaluation offers somewhat partial results, and, as it is important to know to what extent the results carry over collections, we have decided to evaluate the system in its optimal configuration against the test collections of the SENSEVAL-2 competition, with interesting dissimilarities between the all-words task and the lexical sample task. This results complement those obtained with SemCor.

In section 2 we review past work in the area, in section 3 we explain the main algorithm and all the variants. In section 4, we describe the evaluation procedure and the results obtained. Finally, we offer our conclusions in section 5.

Preprint version of the paper published in Expert Systems with Applications, doi:10.1016/j.eswa.2011.01.150

2 Materials and Methods

Some approaches to WSD make use of sources of information other than electronic dictionaries alone. WordNet is one of the most widely used. WordNet is a lexical knowledge base that comprises several semantic relations between their concepts. Among them, the hypernymy relation is usually considered to play an important role in the field of ontologies[1]. There are WSD methods are based on the hypothesis that a fragment of text is about *something*, about a concept or topic more or less specific, so that it induces a preference for some combinations of word senses over others. When that *something* is a concept in a taxonomic hierarchy such as the hypernymy hierarchy in WordNet, then we speak of conceptual-based systems. When it is a topic, as in the case of domain labels associated to WordNet synsets, see for instance [Magnini et al., 2001] then we are dealing with domain-based systems. In both cases the relations are hierarchical, but domain relations and conceptual relations should not be mistaken.

A similar idea is expressed in other words in [Manning and Schütze, 1999] :

*T*he basic inference in thesauri-based disambiguation is that the semantic categories of the words in context determine the category of the text as a unit, and that this category in turn determines what senses of the words are being used.

2.1 Domain relations

One attempt to profit from that hypothesis can be found in [Walker, 1987]. Walker applied the following algorithm. There is a thesaurus in which every word may have several subject codes. The senses of a word with each possible subject code are identified. To disambiguate, he computes a score for each sense counting how many words in the context have the same sense (subject code). The sense with the highest score is chosen.

This algorithm was revisited in [Black, 1988] and had a moderate success since it got a precision around .50 over a sample of five words. The words in question were very polysemic and considered difficult.

In these cases, the hierarchy would be degenerated to a single level; at first sight it might look like a flippant classification, but the truth is that many systems designed later perfectly generalize this approach. It is also true that this approach bears a certain resemblance with the experiment described in [Yarowsky, 1992].

Another system based on hierarchical domain relations is the one described in [Magnini et al., 2001]. A WordNet extension called *WordNet domains* [Magnini and Cavagliá, 2000] is employed. In this extension, each WordNet concept (*synset*) is assigned one or more domain labels. To disambiguate at this domain-level, a formula is used that takes into account the relative frequency of each occurrence of a sense (estimated according to the hand-tagged collection SemCor) belonging to a word in the context. This aspect is important because this sense distribution is usually heavily skewed. The information for each context word is combined so that each domain gets a score. It is easy then to discard the senses for which no domain has enough score and thus disambiguate the words. This allows for a two-fold disambiguation, at the sense level, and at the domain level.

Magnini domains are taken from the *Dewey Decimal Classification*[2], and they are of hierarchical nature, with a three-level distribution. However, for the experiments (the system participated in the SENSEVAL-2 competition) a simpler plain view was used. The case of domain-level disambiguation could be considered as coarse-grained disambiguation. In any case, sense-level precision is very high, crediting the hypothesis of one domain per sentence. A similar, although simpler strategy had already been tested in [Wilks and Stevenson, 1996], where LDOCE [Procter et al., 1978] subject codes where used to disambiguate words

[1] Although ontology experts often dismiss this idea as a common misconception.

[2] http://www.oclc.org/dewey

Preprint version of the paper published in Expert Systems with Applications, doi:10.1016/j.eswa.2011.01.150

after having determined its grammatical category with a part-of-speech tagger, in this case Eric Brill's rule-based tagger presented in [Brill, 1992]. We leave aside domain-based systems here and concentrate in conceptual-based ones.

2.2 WordNet and conceptual density

The hypernymy hierarchy in WordNet has generated a great deal of research with respect to its potential for WSD in general and for conceptual density measures in particular, some of which we revise next.

2.2.1 Voorhees' hoods

The starting idea in [Voorhees, 1993] is that nouns in a sentence have a common topic. The motivating example invites to imagine a sentence with the words *base*, *bat*, *glove* and *hit*. The topic or domain common to all these words would be baseball. With this idea in mind the author proposes an algorithm that marks the synsets in the hierarchy of WordNet (version 1.2 in this case). This algorithm seems to be seminal of the work of [Agirre and Rigau, 1995, Agirre and Rigau, 1996, Montoyo, 2002] among others. We briefly describe the algorithm proposed. The key concept here is that of *hood* of a sense of a word. A hood for a synset s is defined as the biggest connected graph that contains s and only contains descendents of an ancestor of s and does not contain any synset whose descendants include a member of s (as synonym set[3]). The rationale is that the hood for a synset is the area where a word is not ambiguous (this idea is due to Miller). As synsets may have more than one father (i.e. hypernym) they can have several hoods. They might as well not have any. According to the author, this hoods can be used like the categories in other sense inventories like LDOCE or Roget's Thesaurus [Chapman, 1977]. The class indicator of a hood is its root synset.

There is also a general tagging procedure for a noun. Each sense of the noun is selected and its hypernyms visited, noting how many times we visit each node. With this procedure clear, the algorithm consists of the following :

The collection of texts is taken and for each noun the tagging procedure is applied. We call the result global visits. Now, for any one text in the collection, we take each noun we want to disambiguate, and we'll call again the tagging procedure. The result is called now local visits. We now define two concepts for each sense of a particular word in a particular text :

$$\text{Local visit ratio} = \frac{\text{\# local visits to the indicator of the hood of the sense}}{\text{\# local calls to the tagging procedure}}$$

$$\text{Global visit ratio} = \frac{\text{\# global visits to the indicator of the hood of the sense}}{\text{\# global calls to the tagging procedure}}$$

Finally, the score for each sense is :

$$\text{Score} = \text{Local visit ratio} - \text{Global visit ratio}$$

In other words, the difference between the ratio of the times the indicator of the hood has counted for the text and the ratio of the times the indicator of the hood has counted for the collection is computed. Only positive differences are considered. After this process, the sense with the highest score is chosen.

There has not been an exhaustive evaluation of the disambiguation algorithm, it has been used as an intermediate step in Information Retrieval (IR). The conclusion that the author extracts is : *The is-a relation*

[3] Synsets are synonym sets, indicative of different senses of words, for instance {church, building} and {church, Christianity}.

Preprint version of the paper published in Expert Systems with Applications, doi:10.1016/j.eswa.2011.01.150

defines a generalization/specialization hierarchy that is not enough to select the right sense of a noun from the set of fine-grained distinctions in WordNet. This experiments were undertaken with one of the first versions of WordNet. Maybe her conclusions would be slightly different if the experiments were replicated with a more updated version.

2.3 Sussna's minimal distance

In [Sussna, 1993], Sussna faces semantic ambiguity with the intent of trying to improve information retrieval (IR). To defend his stance he cites the classical synonymy and polysemy problems that WSD could help mitigate but there are not actual retrieval experiments, so IR is merely used as an excuse to proceed with WSD.

The first thing to be done is to compute weights for the arcs between WordNet's synsets. All nominal relations (hypernymy, meronymy, holonymy...) are considered and the weight is interpreted as the *cost* of an arc, taking into account that the more arcs of a certain relation go out, the more the meaning of that node is *dispersed*.

Next, the texts to be indexed are processed. The processing consists of running a morphosyntactic tagger, converting the text to small letters, punctuation sign and stop-word obliteration as well as non-noun elimination, together with any surviving word without a nominal entry in WordNet.

With respect to disambiguation, the hypothesis is that if we take a set of nouns that appear close to one another in the text, each of which can have multiple meanings, we can select the correct senses if we choose the combination of senses that minimizes the sum of pairwise distances.

Three alternative strategies are introduced in order to carry out the experiments. In the first one, mutual restriction, every way of selecting one sense of each noun of a predefined set is considered. All the possible pairs for each combination are formed and the average distance is computed for each combination. The *energy* is the minimum of these values, that is, the combination with the lowest average.

The second strategy consists in selecting first, with the previous method, the senses for an initial set of just two nouns and then disambiguate one more noun at a time keeping fixed the senses of the previously disambiguated nouns (Sussna calls this *frozen selections*). Obviously this methods is much less expensive computationally than the first one.

As a third strategy, several variants are considered, such as the frozen strategy when the initial set is a window of a fixed size that is disambiguated with the mutual restriction. In all cases there is a moving window of text that advances one word at a time. In the case of the mutual restriction, it is remarkable that only the central word of the window is tagged, so that in the next window with a new central word, the word previously disambiguated could prefer another sense, thus offering a flexibility which seems more interesting with finer-grained sense inventories. Also, the snowball effect that many NLP algorithms suffer is avoided so that a high amount of failures in an area won't provoke a progressive degradation of the results.

The author distinguishes three cases when it comes to evaluate; when there is one correct sense for a word in context, when there is more than one and when there is no correct or applicable sense. Two measures are defined, one of them would be the near-equivalent of precision over polysemic words and and the other, more complex, tries to capture the difficulty of each particular case. The *gold standard* (the set of annotations considered correct) is manually crafted by the author. To estimate the amount of information available to the algorithm after stripping so many words, the same trimmed text is given to human taggers to annotate in conditions similar to those of the algorithm. They obtained a precision of .78 compared to tagging with the full text, so there seems to be quite a bit of a loss of information even for humans.

The results are consistently superior to a random heuristic. The optimal size of the moving window is found to be 41 words. Theses results confirm that it is better to take into consideration all the relations in

Preprint version of the paper published in Expert Systems with Applications, doi:10.1016/j.eswa.2011.01.150

WordNet and not only hypernymy/hyponymy. On the other hand, the weighting scheme based on outgoing flow does not seem as much influential. Considering the depth of the nodes in the hierarchy (used in calculating the weights) is beneficial. Last, mutual restriction is much better than the frozen heuristic at same window sizes, but for efficiency reasons (the complexity of mutual restriction is clearly exponential) it is not possible to take to experiments very far with the first option.

This work inspired [Agirre and Rigau, 1995,Agirre and Rigau, 1996] among other works. Results corroborate the idea that considering all possible combinations every time is better than *freezing* disambiguations over previous words, thus allowing the meaning of the same occurrence of a word to be different as the context varies. This is probably the most interesting empiric conclusion, especially since direct comparison with other methods is infeasible due to the particular evaluation measures. A high precision was not to be expected since it is an algorithm running on scarce information (disambiguating the reduced texts was difficult even for humans).

2.4 Resnik's contribution

Another interesting contribution is [Resnik, 1998]. The paper starts arguing that it would be interesting to build characteristic vectors for word senses based on word coocurrence features or, even better, on sense coocurrence features. The small amount of sense-tagged data makes not viable the sense coocurrence approach, but the first one is still possible. Among this *distributional* approaches to the problem, Resnik classifies [Hearst and Schütze, 1993], that used the coocurrence counting algorithm described in [Schütze, 1993] which was basically the same as in [Schütze, 1992], and also in [Resnik, 1993].

The author remarks, as will be done also in [Budanitsky and Hirst, 2000], that similarity is a more specialized notion than association or relation. As an example, Resnik argues that medical doctors and nurses might be highly associated, but they are not particularly similar.

A WSD algorithm is sketched out. It is not directed to arbitrary contexts but only to groups of nouns that are already related for some reason, for instance, belonging to the same thesaurus (they are often about specialized fields) or perhaps they have been grouped together by some distributional clustering algorithm. A great deal of the paper is spent in clarifying the differences between the algorithm described and that of [Sussna, 1993]. Resnik's algorithm works this way : First, all the possible pairs of nouns are formed. Then the similarity between every sense of the first noun and every sense of the second is computed according to a measure described in [Resnik, 1995]. Using that same similarity measure, the most informative concept is found, a concept that subsumes both words. If a word sense is a descendant of the most subsuming concept, its score is raised. This is done for all pairs. Finally, the scores are normalized for each word.

Resnik compares this approach with the one in [Lesk, 1986] and especially with the work of Sussna, that we have already reviewed, considering that it is the most similar to his. The algorithm is used to disambiguate words of the same category in Roget's Thesaurus to WordNet senses, that is, exactly the kind of application for which the algorithm was devised. This will allow the author to later link together, in [Resnik, 1999] WordNet categories, Wordsmyth English Dictionary Thesaurus[4] and CETA (Chinese English Dictionary) [Group, 1982].

The experiments described in [Budanitsky and Hirst, 2000] have the goal of spotting out the best similarity measure among several candidates. Resnik's measure was not among the best, so that it is an open question what would happen if the best ranked measure, described in [Jiang and Conrath, 1997], was applied to Resnik's algorithm or even to the one in [Agirre and Rigau, 1996].

An important objection to these algorithms is that their ability to represent knowledge is very limited. Sussna reports that human taggers had serious difficulties annotating the right senses for words after only lemmatized nouns were left over (Sussna wanted to see humans perform in the same conditions as his

[4] http://www.wordsmyth.net

Preprint version of the paper published in Expert Systems with Applications, doi:10.1016/j.eswa.2011.01.150

algorithm). The Agirre and Rigau approach that we comment next also ignores information coming from parts of speech other than nouns, due to WordNet lacking noun-verb relations at that time.

It is also true from a linguistic point of view that relations among the nouns in a sentence are not always clear. The concept of circumstantial complement already explains that a subject and a complement don't have to be particularly related. Compared to the conceptual approach, it is quite possible that a text fragment belonging to a specific domain is a clearer relation, which would partly explain the good results obtained in [Magnini et al., 2001].

2.5 Agirre and Rigau and their influence

The idea of *conceptual density* is known at least since [Wilks et al., 1990]. In [Agirre and Rigau, 1995] semantic relations between concepts (synsets) in WordNet were taken advantage of to define the concept of conceptual density and apply it to WSD over nouns. The way they did it was to a large extent tributary of the ideas in [Sussna, 1993] and [Voorhees, 1993]. In this case the algorithm consists in taking a moving text window around each word to disambiguate and calculate the concept that *dominates* the window in terms of conceptual density. The senses of the word that fall outside the area of influence of the dominating concept are discarded. The most valuable contribution was the measure of conceptual density itself, that was designed to be sensitive to a good number of linguistic features the authors felt relevant. The results were very positive, although the evaluation measures, oriented to number of senses and not number of words as is customary, did not allow a direct comparison. The evaluation set would be considered as being of reduced size by today's standards .

The same authors, in [Agirre and Rigau, 1996], contributed a comparison between their results, obtained for four randomly chosen documents in SemCor and those in [Sussna, 1993] and [Yarowsky, 1992]. They don't hesitate to consider their algorithm superior. We return to this algorithm later, when we describe the improvements we have incorporated into it.

This algorithm has had considerable influence on the WSD literature. It was used in the SENSEVAL-2 competition in the estonian task as explained in [Vider and Kaljurand, 2001]. Also, [Peh and Ng, 1997] describes a reimplementation of the algorithm and a comparison with the heuristic of selecting the first sense in WordNet (WordNet senses are ordered in decreasing number of frequence in SemCor, so it is equivalent to selecting the *a priori* most frequent sense for each word) with the first sense beating the conceptual density. Another algorithm with a strong reminiscence of conceptual density would be the specification marks algorithm presented in [Montoyo et al., 2001, Montoyo and Suárez, 2001, Montoyo, 2002].

The specification marks algorithm solves some of the problems in the original algorithm by Agirre and Rigau. Said algorithm considered each possible sense for a word as a *massive* point in the density calculations. This throws some shadows as we will see later when we describe the shortcomings of the algorithm, because very polysemic words count too much in the density computation. The specification marks approach alleviates the problem to some extent counting only how many different words contribute senses in each subhierarchy. We employ a similar approach developed approximately at the same time in the improved version of the algorithm. Unfortunately, the evaluation of the specification marks algorithm in [Montoyo et al., 2001] uses the same non-standard metric in [Agirre and Rigau, 1995, Agirre and Rigau, 1996]. Anyway, the results of the system in SENSEVAL-2 are available and we compare it to our own approach in the evaluation section. Another well-known problem with Agirre and Rigau's algorithm is that it is not unusual that several senses of a word get in a tie. In order to disambiguate in these cases, Montoyo et al. employed heuristics based on coocurrence of the synonyms conforming WordNet *synsets* and they also incorporated the glosses of the synsets into play. Combining this features of the synsets with hypernyms and hyponyms they succeeded in improving the results of the original algorithm, simplifying at the same time its initial formulation.

Preprint version of the paper published in Expert Systems with Applications, doi:10.1016/j.eswa.2011.01.150

In the field of text clustering, the work in [Hotho et al., 2003] took some inspiration in the work of Agirre & Rigau to determine if disambiguation could contribute to improve results, with a positive outcome.

Another incarnation of conceptual density was also used in the second phase of the WSD algorithm presented in [Mihalcea and Moldovan, 1999, Mihalcea and Moldovan, 2000b]. By way of example, a noun-verb pair would be disambiguated like this : All possible pairs of one sense from each word are formed and conceptual density is computed. The chosen senses are those that maximize conceptual density. The density measure is based this time on counting coocurrences of nouns in the glosses of the concepts in the subhierarchy of the senses of the noun and the verb, together with other intervening factors such as the amount of nouns in common and its depth in the hierarchy (the depth of the synset whose gloss they belong to). The results are superior to the first sense heuristic this time. Mihalcea and Moldovan compare the results with the ones presented in [Stetina et al., 1998]. They also explain how to generalize the algorithm for verb-verb and noun-noun pairs.

Category	Filter	First sense	Final result
Nouns	.76	.80	.87
Verbs	.60	.63	.67
Adjectives	.80	82	.80
Adverbs	.87	.84	.87

Table 1: Final results for Mihalcea and Moldovan

The disambiguation made use of WordNet-1.6 together with the Altavista search engine and the first document in Semcor. The algorithm has two phases. The first one filters senses with Altavista and the second one applies conceptual density to decide between remaining senses. All open-class words are disambiguated and a comparison between the filtering, the first sense heuristic and the final results can be seen in Table 1.

3 Calculation

The basic elements for the algorithm to work are a lexical knowledge base (LKB) with conceptual information (such as Wordnet's *synsets* o synonym term sets), a binary relation R (usually the *is-a* relation in a taxonomy) between the concepts in the LKB, and a formula (see below) that returns the conceptual density of a concept with respect to a certain amount of subconcepts activated through R.

To disambiguate a word the procedure is as follows : First, we take the context surrounding the word and make a window of a fixed given radius. Then we rank the senses of the word in the middle (the word to be disambiguated) of the window following these steps :

- We look up the senses of all the word forms in the window, for each sense of each word we take a number of concepts associated by relation R and weight them according to a formula.
- For each sense of the central word in the window, the related concept (through transitive application of R) with the highest conceptual density gives the score for that sense.
- We normalize the scores for the senses of the word to disambiguate and then take the results as the outcome of the algorithm.

These steps define a template of algorithms of conceptual density with a wide range of possibilities. In the next section we discuss the parameters that we have taken into account and the values with which we have fed them.

Preprint version of the paper published in Expert Systems with Applications, doi:10.1016/j.eswa.2011.01.150

3.1 Parameters

Transitive Relation R The most obvious possibility is perhaps hypernymy, but we have also considered the union of semantic relations such as hypernymy and meronymy (the relation *is-part-of*).

Conceptual density measure We have tried three different measures of conceptual density :

1. The original density formula by Agirre & Rigau [Agirre and Rigau, 1996]:

$$CD(c,m) = \frac{\sum_{i=0}^{m-1} adesc^{i^{\alpha}}}{\sum_{i=0}^{h-1} adesc^{i}} \quad (1)$$

   where *adesc* is the average number of descendents of the concept c, according to transitive use of R, m is the number of marks (or activated synsets) in the subhierarchy of c. Finally, h is the depth of the subhierarchy below c. We have called this formula SAR (Strict Agirre-Rigau). The α factor has been set to 0.2 since this value optimized results for their experiments with WordNet-1.4

2. The same formula without α. We have called it AR (Agirre-Rigau).
3. The LF (logarithmic formula) formula given by :

$$LF(c,m) = \frac{1}{desc_c} log_2 d \sum_{i=0}^{m-1} adesc^{i}$$

   Where $desc_C$ is the number of descendents of c and d is the depth of the concept c in the hierarchy. This formula is similar to the original one with a correction factor that favors more specific items (concepts deeper in the hierarchy). The original formula is supposed to already be sensitive to this aspect but we wanted to give it more chances to influence the final result.

Window size We have tested several window sizes, ranging from 3 to 701 nouns.

Selection of related synsets As far as selecting related concepts is concerned, we have considered two variations.

- In the first place, we have a parameter to eliminate the relations between the higher levels of the hierarchy induced by the transitive closure of R. The reason behind this decision is that higher levels in broad conceptual hierarchies tend to be highly subjective. If there is a concept representative of the topic discussed in the word-window and this concept is to have any effective influence in the disambiguation task, it shouldn't be too abstract or generic (as the concepts in the higher levels usually are) compared to the senses of the words being disambiguated. Conceptual density formulas are designed to reflect this motivation but for broad window sizes, it seems inevitable that the synsets in the top ontology[5] of WordNet will obtain high densities. We represent with a value of zero in this parameter the situation in which the whole hierarchy is considered.

[5] WordNet's top ontology is a reduced set of synsets placed in the summit of the hypernymy hierarchy

Preprint version of the paper published in Expert Systems with Applications, doi:10.1016/j.eswa.2011.01.150

- Second, we introduce another parameter, l, included to consider only the l closest concepts through consecutive application of R. In other words, when we calculate the density of a concept c, we won't consider the density of a concept s related by R if we have to iterate through R more than l times to connect them. This way we effectively restrict the transitivity so that somehow related concepts count, but very distantly related ones don't. Ideally, a concept and its immediate hypernym are be closely semantically related, (as is, for instance, the case with highway$_1$[6]. and road$_1$). Nevertheless, although surely a highway is an entity, it is unlikely that this information will have any impact in the WSD task. It might seem that this parameter and the preceding one are quite similar, however, the results show radically different behavior. The case when unrestricted application of the transitive relation is allowed is represented by a zero value for this parameter.

Synset weighting To compute the conceptual density of a concept c, in the hierarchy induced by R, we have chosen three options to count how many marks or activated synsets m fall below it :

synsets This counting scheme consists of counting each sense of each word in the window that is related with concept c as one mark. This coincides with the formulation of the Agirre-Rigau algorithm. This scheme has a straightforward problem and it is that words interfere severely with themselves. If we take for example the word *end*, which has 14 senses in WordNet[7], and draw the hypernymy hierarchy for the senses below *entity* (with some intermediate nodes omitted for clarity) we obtain the picture in Figure 1

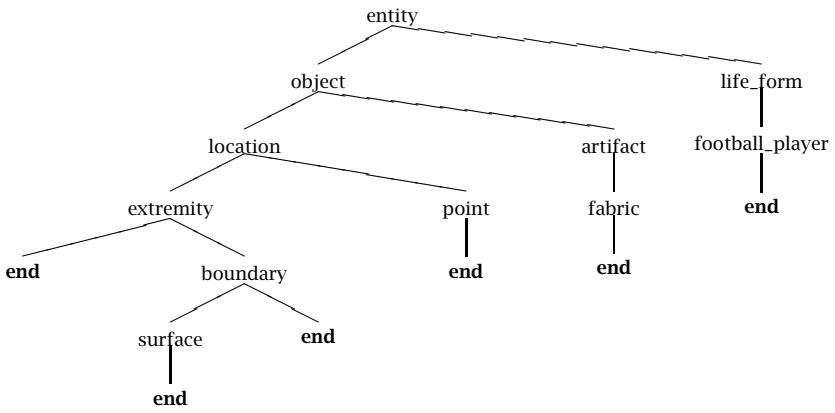

Figure 1: The hierarchy of *end*

It is easy to notice that the other eight senses of end (which are not hyponyms of *entity*) probably will be discriminated with respect to these ones because, in the absence of context, the concept *object* in the figure will get a high density. If we add words to the window context, it is to be expected that the majority of senses will fall in the hierarchy below *entity* (the largest by far) and then the algorithm would discard the other senses. Another adverse effect for highly polysemic words is that they tend to dominate the density measures. For instance, *end* has 14 senses and so 14 massive points to count in the density measures. That seems hardly appropriate if we take into account that around one-third of the words in running text are monosemous and thus

[6] We follow the convention that w_i is the i-th sense of w in WordNet
[7] The example is taken from WordNet-1.5

Preprint version of the paper published in Expert Systems with Applications, doi:10.1016/j.eswa.2011.01.150

contribute only with one mark to the formulas. We have tested two other weighting schemes with the goal of minimizing these effects :

fractional This scheme means weighting each sense of a word in the window as $1/m$, where m is the total number of senses for that word, in order to prevent a highly polysemous word from biasing the conceptual density, although this won't keep some words from being able to disambiguate themselves.

words This is counting as different marks in the subhierarchy of a concept c, only the number of different words contributing with senses under c. This way, all the words in the window contribute to the same extent, independently of their degree of polysemy, and also, a high intra-word density (usually derived from the fine-grainedness of WordNet senses) should not discriminate the senses of that word outside the area. This weighting scheme shares some aspects with the Specification Marks described in [Montoyo et al., 2001] developed independently.

4 Results and Discussion

The parameters of the algorithm have been adjusted to optimize recall measured over SemCor, but we are also interested in finding its higher potential precision. First we present the results over this collection and later we complete the evaluation with the results over the SENSEVAL-2 English tasks collections.

4.1 Evaluation over SemCor

The evaluation has been carried out over the SemCor collection, a set of 187 documents where all content-words have been hand-annotated with the most appropriate sense in WordNet. In our evaluation, each one of the tested configurations of the algorithm has been run over every noun of every document in SemCor and exclusively over nouns, so results cannot be extrapolated to performance in general.

The performance of the algorithm is presented in terms of *precision* and *recall* as defined for the first edition of the SENSEVAL competition :

The scoring system allows scores between 0 and 1, when the system returns more than one sense per word, with the probability mass shared between them. The precision is calculated by dividing the system's score over correct senses by the number of items responded. The recall is calculated dividing the system's score over correct senses by the total number of items to be disambiguated.

4.2 Performance over nouns

Table 2 compares the original Agirre-Rigau algorithm with our best conceptual density system, which we have named ARF, and a baseline : The most frequent sense (which comes defined by the first sense in WordNet for a word, which, in this case cannot be considered as a heuristic since this information is only known after hand-annotating the collection). The precision and recall of the most frequent sense are not the same because in 51 cases the hand-annotation of the correct sense (which includes the lemma of the word) has a wrong lemma. This is the reason why coverage does not reach 1.

We have compared the results of our experiments with random heuristics elsewhere, however, we have decided not to do it this time. The reason is that we consider that a random system should have no prior knowledge over the data. In SemCor, multiword terms (such as *back-and-forth*) which are almost always monosemous, come hand-tagged so that equally sharing the weight of the answer between the senses is relatively good, compared to the results that would obtain a truly random system. In our opinion, in a real situation, multiword-term detection is a merit with a very important impact for a system, a merit that a random system just can't be recognized.

Preprint version of the paper published in Expert Systems with Applications, doi:10.1016/j.eswa.2011.01.150

WSD algorithm	Coverage	Precision	Recall
ARF	.99	.46	.46
Agirre-Rigau	.93	.40	.37
Most frequent sense	.99	.78	.78

Table 2: Performance over nouns (with ties)

Previous work [Gale et al., 1992] supports that the lower bound for a WSD system should be the performance of the most frequent *heuristic*, and the upper bound should be human intertagger agreement. We disagree with Gale et al. since the precision of the most frequent sense (which is not a heuristic) over the nouns in SemCor is .78, a very high figure that looks much more like an upper bound than what we would expect to be a lower bound. SemCor creators estimated an error rate of about 10% in the hand-tagging process. We don't have knowledge of any systems placed in that recall bracket (between .78 and .90). In fact, .78 is the figure that human taggers attained when they were given the input to Sussna's algorithm so it should be an upper bound for his algorithm.

It is also worthwhile to mention another difference with respect to the original algorithm. We have decided to agglutinate all the weight of an answer to the sense with the highest conceptual density. We have done this because we believe it is beneficial with regard to the evaluation measures, as it prevents the system from spreading the weight over too many senses. The original algorithm was presented prior to these standard measures, which might influence the comparison.

The results presented in [Agirre and Rigau, 1996] were more promising than those obtained in our recreation of their algorithm, but they were obtained over a test collection nearly 50 times smaller (they only used 4 SemCor documents out of 187). Besides, their definitions of precision, recall and coverage were different and not easily adaptable.

Our system achieves .46 recall, a relative improvement of 24% over Agirre-Rigau's system. This last algorithm uses a window size of 35, which explains why the coverage is slightly lower than that achieved by our system (which uses a window size of 271). It is a considerable improvement over the departure point but results are still much poorer than the most frequent baseline.

A direct comparison with this baseline could lead to discard conceptual relations as a source for WSD. This would be, however, a mistake for several reasons :

- Manual annotations, taken as the gold standard, are biased in favor of the first sense in WordNet, that corresponds by construction with the most frequent sense. Human annotators, in an all-words task, have to select the most appropriate sense for a different word every time. Each polysemous word in running text has an average of five senses. Inevitably, the annotator tends to select the first sense that seems to fit in the context, and this produces a bias in favor of the first senses. Studies about WSD evaluation [Resnik and Yarowsky, 1999, Kilgarriff and Rosenzweig, 2000] speak in favor of an annotation task over a lexical sample of occurrences of selected words, in which the annotator tags repeatedly occurrences of the same word, thus reaching a certain familiarity with the senses of the chosen word. This was accomplished in SENSEVAL-2 lexical sample task. In that task, the first sense heuristic was beaten by many systems in stark contrast to the all-words task in which only three systems succeeded in beating the first sense.

- Leaving aside the problems of hand annotation, the all-words task implies that the system must try to repeatedly disambiguate occurrences of very common words, that can have twenty different senses

Preprint version of the paper published in Expert Systems with Applications, doi:10.1016/j.eswa.2011.01.150

or more is the LKB [8]. This terms are almost impossible to disambiguate, and probably its correct disambiguation is nearly useless for most applications.

- A comparison with the most frequent sense (for nouns) should not be done without realizing also that it is very close to human intertagger agreement, which is what is considered optimal disambiguation, since it is unclear what interpretation could be done of a system that performed a human task better than humans. In the case of SemCor as we have previously outlined, choosing the most frequent sense is much more than a supervised heuristic.
- Our results suggest that the algorithm is likely to be very efficient for the kind of task for which it was created; the disambiguation of the genus term in dictionary entries. These are often comprised in the case of nouns of one or two sentences that describe the sense of the noun in question making a explicit reference to its generic (*genus term*) and the differences (*differentia*) with respect to it. In order for this definition to be of any use, the genus is usually a hypernym, and not a far away one (A sentence like *A banana is a fruit* is more informative than *A banana is an entity*. Disambiguating within the sentence, and limiting the exploration levels (parameter l in earlier description) to three, the situation should be very much like that of a dictionary entry for a noun. Here we have limited ourselves to disambiguating running text, but given the high precision achieved (.65, as we will show in the following subsection) we think that the algorithm would be very efficient in genus term disambiguation in dictionary entries. If recall is not higher, it is in fact because the presence of close hypernyms is not usual in running text. Also because this algorithm is not able to discriminate between two senses in the hierarchy that are brothers (sons of the same father) according to R, since they would get the same density. In the case of WordNet and the hypernymy relation this situation is very common.

A more appropriate conclusion would then be that conceptual measures alone are not sufficient to carry out a precise WSD. In this point we agree with the conclusions in [Voorhees, 1993].

Another improvement that we introduced *a posteriori* consists in deleting the answers where all the senses are in a tie. This is because when several senses of the word are siblings, the parents in the hierarchy systematically obtain the same score. This is a shortcoming of the algorithm and many senses suffer from this problem in WordNet.

A possibly better solution to this problem might be using coocurrence heuristics such as the ones in [Montoyo, 2002], in order to break the ties between senses in the specification marks method.

The results after this little correction have increased precision and decreased recall as expected. These are the new values : The algorithm disambiguates 72896 test cases (82%), with precision .50 (from .46 before) and recall .41 (from .46). All the evaluation data onwards has been computed eliminating the answers with ties.

We proceed now to the discussion of the detailed influence of every parametererized factor.

4.3 Window size

Figure 2 shows the behavior of the algorithm with window sizes that range between 1 and 701 nouns.

The average number of nouns per document is 422 and the maximum is 649. We have tested moving windows until size 701. Why is the graphic not constant between size 651 and 701? The answer is in the way the window is built in document borders. A window of size 51 would, in general, be a window with a central word and 25 words at each side. This is not possible in document borders : The first noun doesn't have any context to the left and we have the analogue problem with the last one. We have chosen to fit

[8] It is not uncommon that systems are programmed to try not to disambiguate very common and polysemous words such as the different forms of *be*, *have* and *do*

Preprint version of the paper published in Expert Systems with Applications, doi:10.1016/j.eswa.2011.01.150

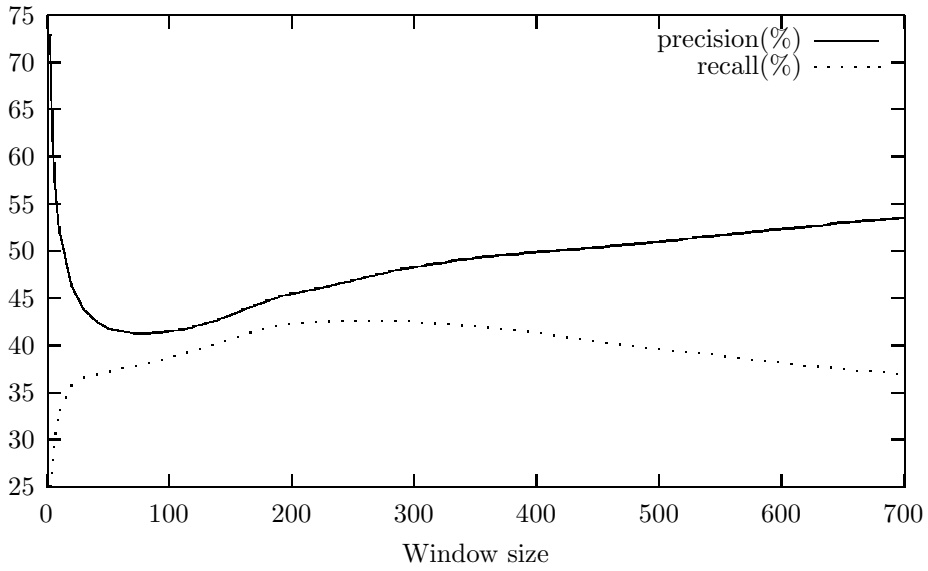

Figure 2: Window size effect

the *side context*. For the window in the aforementioned example we would choose the first time the first noun and the next 25 as right context. The following time (the window moves one noun at a time in order to process the whole document) the central word will the second noun and we will have one noun as left context and 25 as right context. Then we would take a left context of 2 nouns and a right context of 25 and so forth. The same idea applies to the end of the document, so that whenever it is impossible to have the same amount of context to each side at least there's some balance between them.

The way to get the whole picture would then be to take the window size until 1299, so that the first window already spreads through the whole document in all cases. That option was too resource-consuming and the differences in the graph are increasingly smaller with window size.

In contrast to what we expected, recall does not stabilize with big window sizes. Now that we have discarded ties (pseudo-random answers) we have learned that recall goes down again with long window sizes, reaching an optimal value with a width of 271 nouns.

Precision reaches its peak with a three word window, with very low coverage though, as can be assessed from the broad gap in the figure between precision and recall at that point. With respect to limiting applications of R to find related concepts, if we only allow one application, results are .28 coverage, .74 precision and .21 recall.

This has caught our attention so that we have also experimented what happens when the disambiguation context is the current sentence. The algorithm disambiguates 33600 nouns (38%) with precision .65 and recall .24.

It is convenient to take into account that the percentage of monosemous expressions (nouns and multiword terms in WordNet tagged as nouns in SemCor) is high : 20%.

4.4 Type of conceptual relation

Table 3 shows the results of algorithm runs with different semantic relations. Apparently, meronymy/holonymy relations do not add useful information up to hypernymy, although they achieve remarkable precision by

Preprint version of the paper published in Expert Systems with Applications, doi:10.1016/j.eswa.2011.01.150

themselves (with a low recall due to the low coverage).

Relation	Precision	Recall
Hipernymy	.50	.41
Hipernymy + Meronymy	.50	.41
Hipernymy + Holonymy	.50	.41
Meronymy	.61	.25
Holonymy	.61	.25

Table 3: Precision and recall with different conceptual relations

4.5 Conceptual density formula

System	Attempted	Coverage	Precision	Recall
AR	72896/88058	.83	.50	.41
LF	84685/88058	.96	.39	.37
ARS	84658/88058	.96	.36	.35

Table 4: Effect of conceptual density measures

The effects of the density formula can be seen in Table 4. LF formula performs worse than the original Agirre and Rigau formula. On the other hand, the α parameter, that was adjusted to 0.2 with the intention of optimizing disambiguation over four specific documents in SemCor1.4, is clearly inadequate to evaluate against all the SemCor documents in WordNet-1.7: α=1 (AR) produces a relative improvement of 19% over α=0.2 (SAR)

The different formulas yield recall figures between .35 and .41, showing that choosing an adequate formula does have an impact on the results. Maybe a more adequate formula could improve results even further.

4.6 Synset selection

Elimination of upper levels

Figure 3 shows the effect of eliminating the relations between the synsets in the upper levels of the hierarchy. Contrary to our presuppositions, eliminating the relation only in the upper two levels already affects negatively the results. Deleting more than six levels produces virtually random behavior, since most of the nominal information in WordNet is encoded in those six levels.

Limiting iterative application of relation

The effect of limiting the chains of hypernyms is shown in Figure 4. The graph shows that the algorithm is useless without this constraint and the optimal limit is three.

This criterion confirms that going up in the hierarchy without restrictions introduces noise, due to very generic concepts that spoil the performance.

These results seem to indicate that this WSD algorithm is not behaving as expected : The upper levels participate actively in the disambiguation and so, the average conceptual density is using concepts too

Preprint version of the paper published in Expert Systems with Applications, doi:10.1016/j.eswa.2011.01.150

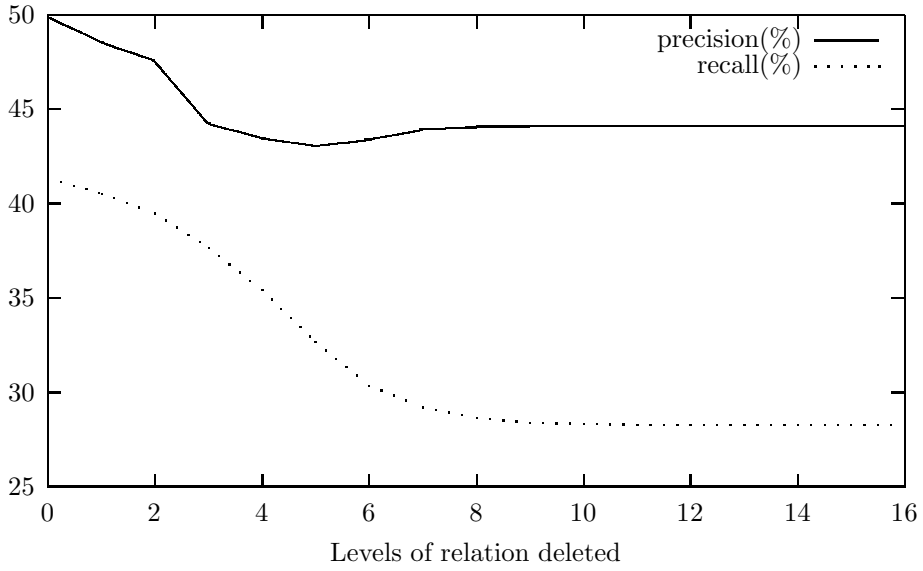

Figure 3: Effect of deleting relations in the upper levels

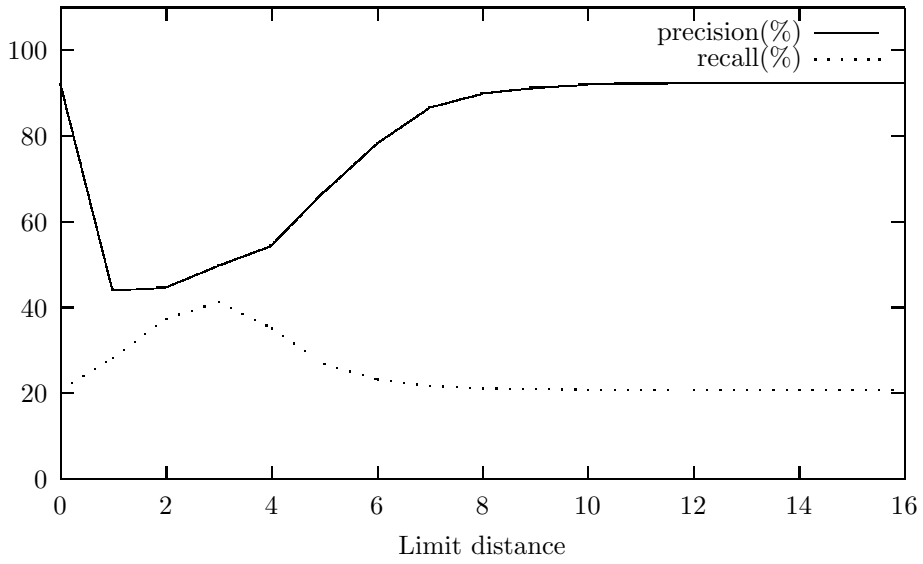

Figure 4: Effect of limiting length of inspected hypernymy chains

Preprint version of the paper published in Expert Systems with Applications, doi:10.1016/j.eswa.2011.01.150

general to be significative for disambiguation. This can partly explain why density measures don't produce better results.

4.7 Weighting schemes

Table 5 shows recall for the three approaches to sense weighting. Surprisingly, penalizing very ambiguous words according to their polysemy level (the *fractional* approach) doesn't improve performance over the standard scheme (*synsets*). Taking the number of different words contributing senses (*words*) is almost as accurate as the *synsets* scheme but with a considerably higher recall.

Scheme	Precision	Recall
Words	.50	.41
Fractional	.41	.39
Synsets	.52	.30

Table 5: Effect of weighting scheme

4.8 Behavior over different text categories

SemCor documents, a fraction of the Brown Corpus [Francis and Kucera, 1982] are classified according to a predefined set of domains (Press, Fiction, Romance & Love story, Humor, etc...). It is interesting to note how WSD performance changes with the different categories. In Table 6, general performance is broken down according to those categories. Categories where conceptual density works best are placed near the beginning of the table.

Category textual	Polysemy	Precision	Recall
A. Press : reportage	5.74	.53	.36
F. Popular Lore	5.38	.53	.47
B. Press : editorial	5.75	.52	.45
J. Learned	5.31	.52	.46
H. Miscellaneous	5.29	.50	.45
R. Humor	5.81	.50	.44
E. Skills & hobbies	5.61	.50	.46
G. Belles lettres, biographies, essays	5.58	.49	.42
D. Religion	5.59	.49	.42
P. Romance & love story	6.68	.48	.38
K. General Fiction	6.37	.47	.39
L. Mystery & detective fiction	6.53	.46	.34
C. Press: reviews	5.44	.46	.38
M. Science fiction	5.91	.45	.37
N. Adventure & western fiction	6.67	.45	.37

Table 6: WSD performance on different text categories

While the average polysemy does not change excessively with document category, the WSD system works better over non-fictional categories (*Press : reportage, Editorial, Popular Lore & Miscellaneous*, etc...)

Preprint version of the paper published in Expert Systems with Applications, doi:10.1016/j.eswa.2011.01.150

and worse over fictional ones (*Mystery & detective fiction, Adventure & western fiction*). This seems to corroborate the hypothesis that WSD has more applicability in technical documents, where word senses have clearer distinctions, metaphors are less common and the context provides more accurate domain information than in fictional texts.

4.9 Evaluation over SENSEVAL collections

In this case we have decided, instead of optimizing for recall, to look for a compromise between precision and recall and so we have maximized precision within a sentence bounded context.

In order to be able to compare our system (ARF) with the other ones, we have had to re-evaluate all the participating systems in SENSEVAL which have publicly disclosed their results in order to obtain the results for nouns only. The results for the all-words task can be seen in Figure 5. Our system, ARF, is located around the middle of the table, between david_fa_UNED-AW-U and david_fa_UNED-AW-T. We have abbreviated some systems names for clarity.

For the lexical sample task the results are shown in Figure 6. ARF system is the third-worst. We have eliminated some systems variants for clarity. Although the improvement over a random heuristic is important, it is obvious that performance over these highly-ambiguous words is not very good. It ought to be said, though, that while for the all-words task, the supervised systems had to rely on examples adapted from other collections,[9] where few examples could be found for any particular word, for the lexical-sample task the organization released a training collection so that supervised systems had large amounts of tagged examples for each word. Given that the vast majority of participating systems were supervised, it is only obvious that the task was easier for them since very straightforward supervised baselines attain good results[10]

It would be an interesting future work to research if the words for which the algorithm has shown worse performance could be better disambiguated using coocurrence information. That being case would indicate the existence of words for which the correct senses would be more dependent on domain than hierarchy. It is well known that WordNet's hierarchy is a taxonomic classification that does not associate semantic domains. For instance, *tennis_racket* and *tennis_shoe* belong to the tennis domain but are not linked in WordNet (at least in version 1.7 and previous). This kind of link would be of a semantic more than taxonomic nature.

5 Conclusions

We have presented an exhaustive evaluation of a set of several different WSD algorithms that rely solely on conceptual relations among word senses. Our starting point has been the conceptual density algorithm in [Agirre and Rigau, 1995] based on a density measure defined over the nominal hierarchy induced by the hypernymy relation in WordNet. This algorithm, that had a competitive performance over a smaller collection, turned out to be behave worse in a complete evaluation of SemCor. We have experimented several modifications and adjusted the associated parameters. obtaining results for more than one hundred variants of the algorithm, including one which is identical to the original one.

The main conclusions we have reached to are :

- Our system behaves a relative 24% better as far as recall is concerned that the original algorithm. This improvement has been obtained with a linear-complexity implementation

[9] Chiefly SemCor-1.6, since SemCor-1.7 was not ready at the time of the competition and WordNet-1.7 was the sense inventory chosen.

[10] The results of these baselines can be found at www.senseval.org.

Preprint version of the paper published in Expert Systems with Applications, doi:10.1016/j.eswa.2011.01.150

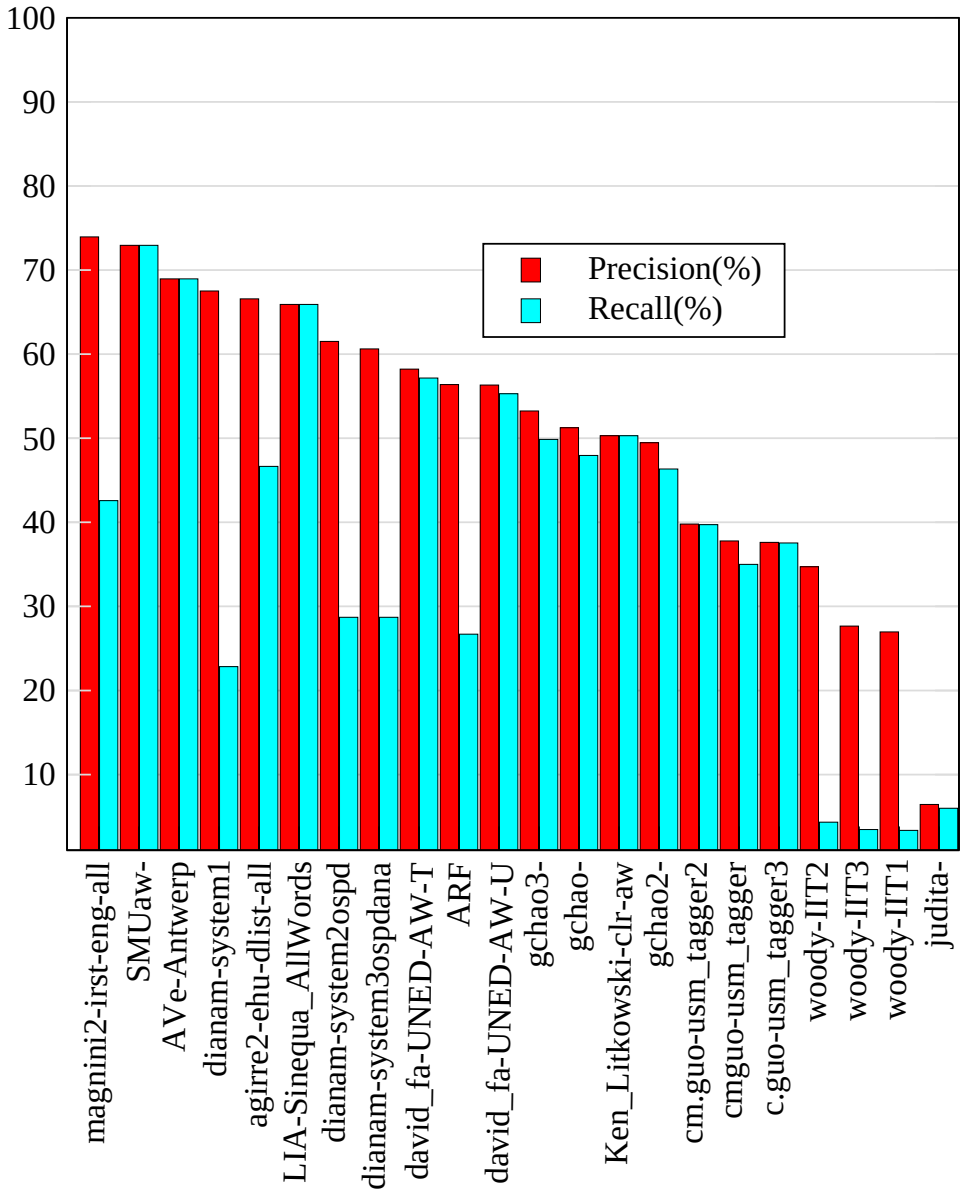

Figure 5: All-words task systems comparison

Preprint version of the paper published in Expert Systems with Applications, doi:10.1016/j.eswa.2011.01.150

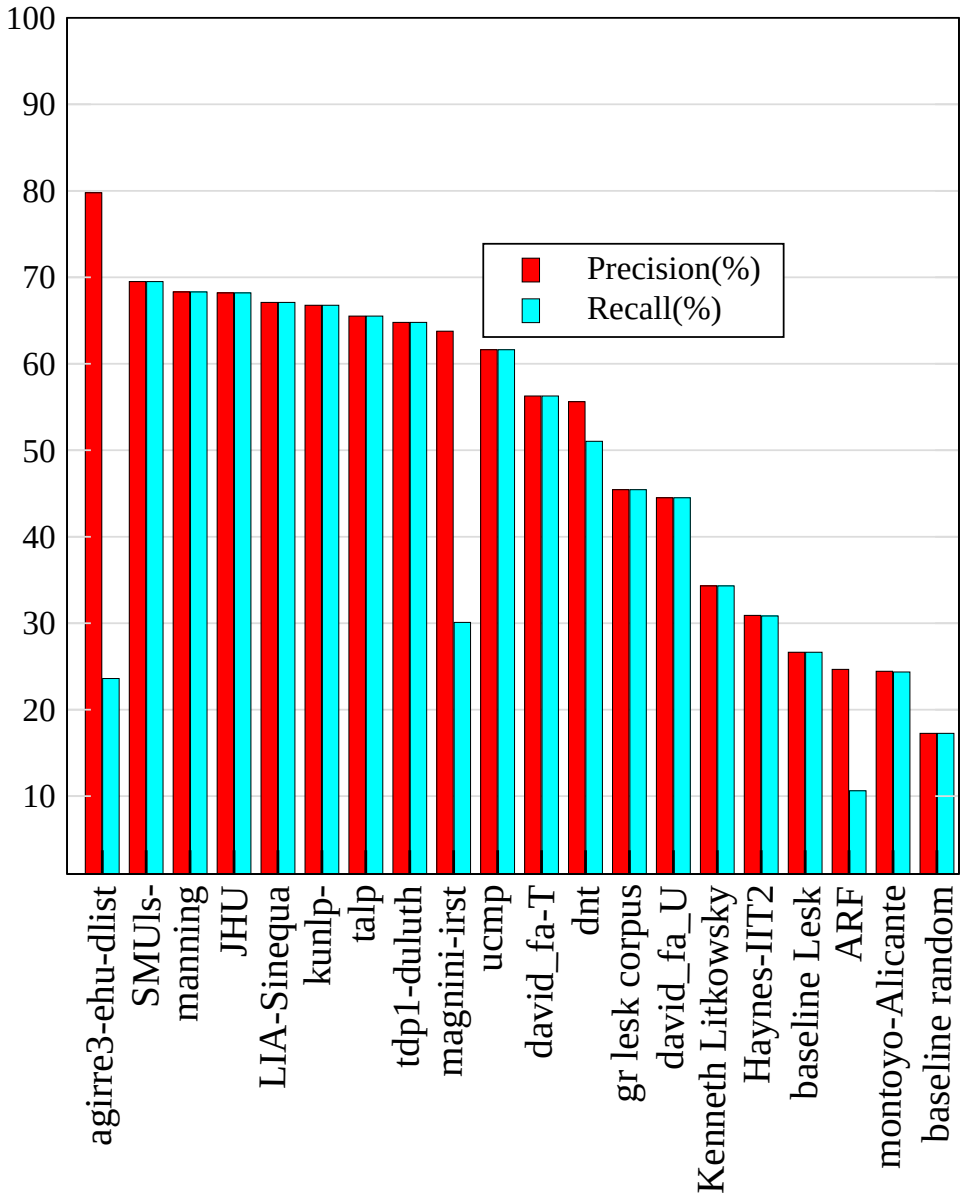

Figure 6: Lexical-sample task systems comparison

Preprint version of the paper published in Expert Systems with Applications, doi:10.1016/j.eswa.2011.01.150

- We have shown that, in practice, the original algorithm used long hierarchical chains to disambiguate, which are related to vague conceptual associations that give noisy results. Our ideal configuration uses hypernymy chains of length three at most, combined with other optimizations in order to keep a good coverage.
- Regarding the context length, things are clear as far as recall is concerned. It reaches a peak at a window size of 271, that is, broad context has useful information to disambiguation. This is in accordance with [Gale et al., 1993]. However, precision behaves somewhat oddly, it is very high for a window size of three (that is, when the first noun to the left and/or right is a hypernym/hyponym or a monosemous word) a relatively uncommon situation (coverage 34%, precision .73, recall .25). Beyond that, precision quickly drops, with a minimum in size 81 of .41 and then again it gets higher, not regaining initial levels, though. A possible explanation for this is that short documents in SemCor are comprised of various related but independent pieces (in the case of news, it is frequent that several short pieces of news make up one document), while longer documents tend to be more homogeneous (such as documents which are novel extracts).
- Surprising behavior in terms of precision of the meronymy and holonymy relations by themselves seems to support a hypothesis according to which an effective method for WSD could consist in detecting a multitude of small linguistic phenomena, relatively scarce but highly accurate and, in the case of conflict, study the way to combine the information.
- We have shown that WSD seems to work better on non-fiction, domain specific texts, than in fiction texts with this technique.

The performance of conceptual relations is relatively low in terms of recall, indicating that this relations should be combined with other types of information (coocurrence statistics, domain information, etc...). The interaction of these knowledge sources is an open field to explore.

Preprint version of the paper published in Expert Systems with Applications, doi:10.1016/j.eswa.2011.01.150

References

[Agirre and Martinez, 2001] Agirre, E. and Martinez, D. (2001). Knowledge sources for word sense disambiguation. *Lecture Notes in Computer Science*, 2166.

[Agirre and Rigau, 1995] Agirre, E. and Rigau, G. (1995). A Proposal for Word Sense Disambiguation using Conceptual Distance. In *Proceedings of the First International Conference on Recent Advances in Natural Language Processing. – Tzigov Chark, Bulgaria, September*.

[Agirre and Rigau, 1996] Agirre, E. and Rigau, G. (1996). Word sense disambiguation using conceptual density. In *Proceedings of the International Conference in Computational Linguistics COLING'96, Copenhagen, Denmark.*, pages 16–22.

[Banerjee and Pedersen, 2002] Banerjee, S. and Pedersen, T. (2002). An adapted lesk algorithm for word sense disambiguation using wordnet. In Gelbukh, A., editor, *Computational Linguistics and Intelligent Text Processing*, volume 2276 of *Lecture Notes in Computer Science*, pages 117–171. Springer Berlin / Heidelberg.

[Black, 1988] Black, E. (1988). An experiment in computational discrimination of English word senses. *IBM Journal of Research and Development*, 32:185–194.

[Brill, 1992] Brill, E. (1992). A simple rule-based part-of-speech tagger. In *Proceedings of ANLP-92, 3rd Conference on Applied Natural Language Processing*, pages 152–155, Trento, IT.

[Budanitsky and Hirst, 2000] Budanitsky, A. and Hirst, G. (2000). Semantic Distance in WordNet : An experimental, application-oriented evaluation of five measures. In *Workshop on WordNet and Other Lexical Resources, in the North American Chapter of the Association for Computational Linguistics (NAACL-2000), Pittsburgh, PA*.

[Chapman, 1977] Chapman, R. (1977). *Roget's International Thesaurus,* 4^{th} *Edition*. Harper and Row, New York.

[Chodorow et al., 2000] Chodorow, M., Leacock, C., and Miller, G. (2000). A Topical/Local Classifier for WSD. *Computers and the Humanities*, 34:115–120.

[Cucchiarelli et al., 2000] Cucchiarelli, A., Faggioli, E., and Velardi, P. (2000). Will Very Large Corpora Play For Semantic Disambiguation The Role That Massive Computing Is Playing For Other AI-Hard Problems? In *Proceedings of the* 2^{nd} *International Conference on Language Resources & Evaluation (LREC)*.

[Dini et al., 1998] Dini, L., Vittorio Di Tomaso, and Segond, F. (1998). Error driven word sense disambiguation. In Boitet, C. and Whitelock, P., editors, *Proceedings of the Thirty-Sixth Annual Meeting of the Association for Computational Linguistics and Seventeenth International Conference on Computational Linguistics*, pages 320–321, San Francisco, California. Morgan Kaufmann Publishers.

[Dorr and Jones, 1996] Dorr, B. and Jones, D. (1996). Role of word-sense disambiguation in lexical acquisition. Predicting Semantics from Syntactic Cues. In *Proceedings of International Conference in Computational Linguistics (COLING), Copenhagen.*

[Fellbaum et al., 1997] Fellbaum, C., Joachim, G., and Landes, S. (1997). Analysis of a Hand-Tagging Task. In *Proceedings of the Applied Natural Language Processing (ANLP) Workshop on Tagging Text with Lexical Semantics, Why, What and How?*, Washington D.C., USA.

Preprint version of the paper published in Expert Systems with Applications, doi:10.1016/j.eswa.2011.01.150

[Francis and Kucera, 1967] Francis, S. and Kucera, H. (1967). Computational Analysis of present-day American English. *Providence, Rhode Island: Brown University Press*.

[Francis and Kucera, 1982] Francis, W. and Kucera, H. (1982). *Frequency Analysis of English Usage: Lexicon and Grammar*. Houghton Mifflin Company, Boston.

[Gale et al., 1992] Gale, W. A., Church, K. W., and Yarowsky, D. (1992). Estimating upper and lower bounds on the performance of word-sense disambiguation programs. *Proceedings of the 30th Annual Meeting of the Association for Computational Linguistics*.

[Gale et al., 1993] Gale, W. A., Church, K. W., and Yarowsky, D. (1993). A method for disambiguating word senses in a large corpus. *Computers and the Humanities*, 26(5):415–439.

[Group, 1982] Group, C.-E. T. (1982). *Chinese Dictionaries : An extensive Bibliography of Dictionaries in Chinese and Other Languages*. Greenwood Publishing.

[Haynes, 2001] Haynes, S. (2001). Semantic tagging using wordnet examples. In Yarowsky, D. and Preiss, J., editors, *Proceedings of the Second SENSEVAL Workshop*, pages 79–82.

[Hearst and Schütze, 1993] Hearst, M. A. and Schütze, H. (1993). Customizing a Lexicon to Better Suit a Computational Task. In *Proceedings of ACL SIGLEX Workshop, Acquisition of Lexical Knowledge from Text*.

[Hotho et al., 2003] Hotho, A., Staab, S., and Stumme, G. (2003). Wordnet improves text document clustering. In *Proceedings of the Semantic Web Workshop at SIGIR-2003, 26th Annual International ACM SIGIR Conference, July 28-August 1, 2003, Toronto, Canada*.

[Jiang and Conrath, 1997] Jiang, J. J. and Conrath, D. W. (1997). Semantic similarity based on corpus statistics and lexical taxonomy. In *Proceedings of the Conference on Research in Computational Linguistics*, Taiwan.

[Kilgarriff and Rosenzweig, 2000] Kilgarriff, A. and Rosenzweig, J. (2000). Framework and results for English SENSEVAL. *Computers and the Humanities*, 34(1-2).

[Krovetz, 1998] Krovetz, R. (1998). More than one sense per discourse. Technical report, NEC Princeton New Jersey Labs. Research Memorandum.

[Kwong, 2001] Kwong, O. (2001). Word Sense Disambiguation with an Integrated Lexical Resource. In *Proceedings of the Workshop WordNet and Other Lexical Resources, of the Second Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL)*.

[Lesk, 1986] Lesk, M. (1986). Automatic Sense Disambiguation using Machine Readable Dictionaries: How to Tell a Pine Cone from An Ice Cream Cone. In *Proceedings of the 5th annual international conference on Systems documentation*, pages 24–26. ACM Press.

[Lin, 1997] Lin, D. (1997). Using syntactic dependency as local context to resolve word sense ambiguity. In *Meeting of the Association for Computational Linguistics*, pages 64–71.

[Magnini and Cavagliá, 2000] Magnini, B. and Cavagliá, G. (2000). Integrating Subject Field Codes into WordNet. In *Proceedings of the* 2^{nd} *International Conference on Language Resources & Evaluation (LREC), Athens*.

[Magnini et al., 2001] Magnini, B., Strapparava, C., Pezzulo, G., and Gliozzo, A. (2001). Using Domain Information for Word Sense Disambiguation. In *Second International Workshop on Evaluating Word Sense Disambiguation Systems (SENSEVAL), Toulouse*, pages 111–114.

[Manning and Schütze, 1999] Manning, C. and Schütze, H. (1999). *Foundations of Statistical Natural Language Processing*. MIT Press, Cambridge, MA.

[Mihalcea and Moldovan, 1999] Mihalcea, R. and Moldovan, D. (1999). A Method for Word Sense Disambiguation of Unrestricted Text. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics,Maryland, NY*.

[Mihalcea and Moldovan, 2000a] Mihalcea, R. and Moldovan, D. (2000a). An Iterative Approach to Word Sense Disambiguation. In *Proceedings of FLAIRS*, pages 219–223.

[Mihalcea and Moldovan, 2000b] Mihalcea, R. and Moldovan, D. (2000b). Using WordNet and Lexical Operators to Improve Internet Searches. *IEEE Internet Computing 4*, 1:34–43.

[Miller, 1995] Miller, G. A. (1995). WordNet: A Lexical Database for English. *Communications of the ACM*, 38(11):39–41.

[Montoyo, 2002] Montoyo, A. (2002). *Desambiguación léxica mediante marcas de especificidad*. PhD thesis, Universidad de Alicante.

[Montoyo et al., 2001] Montoyo, A., Palomar, M., and Rigau, G. (2001). Lexical Enrichment of WordNet with Classification Systems Using Specification Marks Method. In *Proceedings of the NLDB'01*, pages 109–119.

[Montoyo and Suárez, 2001] Montoyo, A. and Suárez, A. (2001). The University of Alicante Word Sense Disambiguation System. In Yarowsky, D. and Preiss, J., editors, *Second International Workshop on Evaluating Word Sense Disambiguation Systems (SENSEVAL), Toulouse*, pages 131–134.

[Moon, 2000] Moon, R. (2000). Lexicography and Disambiguation : The Size of the Problem. In *Computers and the Humanities.*, volume 34, pages 99–102. Kluwer Academic Publishers.

[Ng, 1997] Ng, H. (1997). Getting serious about word sense disambiguation. In *Tagging Text with Lexical Semantics: Why, What, and How? Workshop, Washington, D.C.*

[Ng and Lee, 1996] Ng, H. T. and Lee, H. B. (1996). Integrating Multiple Knowledge Sources to Disambiguate Word Sense: An Exemplar-Based Approach. In Joshi, A. and Palmer, M., editors, *Proceedings of the Thirty-Fourth Annual Meeting of the Association for Computational Linguistics*, pages 40–47, San Francisco. Morgan Kaufmann Publishers.

[Ng and Zelle, 1997] Ng, H. T. and Zelle, J. (1997). Corpus-Based Approaches to Semantic Interpretation in Natural Language Processing. In *Artificial Intelligence Magazine, Special Issue on Natural Language Processing*, volume 18(4), pages 45–64. American Association for Artificial Intelligence.

[Patwardhan et al., 2003] Patwardhan, S., Banerjee, S., and Pedersen, T. (2003). Using measures of semantic relatedness for word sense disambiguation. In Gelbukh, A., editor, *Computational Linguistics and Intelligent Text Processing*, volume 2588 of *Lecture Notes in Computer Science*, pages 241–257. Springer Berlin / Heidelberg.

[Peh and Ng, 1997] Peh, L. S. and Ng, H. T. (1997). Domain-Specific Semantic Class Disambiguation Using WordNet. In *Proceedings of the Fifth Workshop on Very Large Corpora, Beijing*, pages 55–64.

[Procter et al., 1978] Procter, P., Ilson, R., and Ayto, J. (1978). *Longman Dictionary of Contemporary English.* Longman Group Limited, Harlow, UK.

[Resnik, 1993] Resnik, P. (1993). *A Class-based Approach to Lexical Relationships*. PhD thesis, University of Pennsylvania.

Preprint version of the paper published in Expert Systems with Applications, doi:10.1016/j.eswa.2011.01.150

[Resnik, 1995] Resnik, P. (1995). Using information content to evaluate semantic similarity in a taxonomy. In *Proceedings of the IJCAI*, pages 448–453.

[Resnik, 1998] Resnik, P. (1998). Disambiguating noun groupings with respect to wordnet senses. In *Proceedings of the third Workshop on Very Large Corpora, MIT*, pages 95–130.

[Resnik, 1999] Resnik, P. (1999). Semantic Similarity in a Taxonomy: An Information-Based Measure and its Application to Problems of Ambiguity in Natural Language. *Journal of Artificial Intelligence Research*, 11:95–130.

[Resnik and Yarowsky, 1999] Resnik, P. and Yarowsky, D. (1999). Distinguishing Systems and Distinguishing Senses: New Evaluation Methods for Word Sense Disambiguation. In *Journal of Natural Language Engineering*, volume 5(2), pages 113–134.

[Rigau et al., 1997] Rigau, G., Atserias, J., and Agirre, E. (1997). Combining Unsupervised Lexical Knowledge Methods for Word Sense Disambiguation. In Cohen, P. R. and Wahlster, W., editors, *Proceedings of the Thirty-Fifth Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics (ACL-EACL)*, pages 48–55, Somerset, New Jersey.

[Schütze, 1993] Schütze, H. (1993). Word space. In Hanson, S., Cowan, J., and Giles, C., editors, *Advances in Neural Information Processing Systems 5*. Morgan Kaufmann Publishers.

[Schütze, 1992] Schütze, H. (1992). Dimensions of meaning. In *Proceedings of the ACM/IEEE Conference on Supercomputing*, pages 787–796. IEEE Computer Society Press.

[Stetina et al., 1998] Stetina, J., Kurohashi, S., and Nagao, M. (1998). General Word Sense Disambiguation Method Based on A Full Sentential Context. In Harabagiu, S., editor, *Use of WordNet in Natural Language Processing Systems: Proceedings of the Conference*, pages 1–8. Association for Computational Linguistics, Somerset, New Jersey.

[Stevenson and Wilks, 1999] Stevenson, M. and Wilks, Y. (1999). Combining Weak Knowledge Sources for Sense Disambiguation. In *IJCAI*, pages 884–889.

[Sussna, 1993] Sussna, M. (1993). Word Sense Disambiguation for Free-text Indexing Using a Massive Semantic Network. In *Proceedings of the Second International Conference on Information and Knowledge Management (CIKM)*, pages 67–74.

[Vider and Kaljurand, 2001] Vider, K. and Kaljurand, K. (2001). Automatic WSD : Does it make sense for Estonian? In *Second International Workshop on Evaluating Word Sense Disambiguation Systems (SENSEVAL), Toulouse*, pages 159–162.

[Voorhees, 1993] Voorhees, E. M. (1993). Using WordNet to Disambiguate Word Senses for Text Retrieval. In *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 171–180. ACM Press.

[Vossen, 1998] Vossen, P. (1998). *EuroWordNet: a multilingual database with lexical semantic networks*. Kluwer Academic Publishers.

[Walker, 1987] Walker, D. E. (1987). Knowledge Resource Tools for Accessing Large Text Files. *Machine Translation : Theoretical and Methodological Issues*, pages 247–261.

[Wilks et al., 1990] Wilks, Y., Fass, D., Guo, C., McDonald, J., Plate, T., and Slator, B. (1990). Providing Machine Tractable Dictionary Tools. In *Machine Translation 5(2), 99-151.*

Preprint version of the paper published in Expert Systems with Applications, doi:10.1016/j.eswa.2011.01.150

[Wilks and Stevenson, 1996] Wilks, Y. and Stevenson, M. (1996). The Grammar of Sense : Is word sense tagging much more than part-of-speech tagging? Technical report, University of Sheffield, UK.

[Wilks and Stevenson, 1998] Wilks, Y. and Stevenson, M. (1998). Word sense disambiguation using optimised combinations of knowledge sources. *Proceedings of COLING (International Conference in computational linguistics) - ACL (Association for Computational Linguistics) in Montreal, Quebec, Canada*.

[Yarowsky, 1992] Yarowsky, D. (1992). Word-Sense Disambiguation using Statistical Models of Roget's Categories Trained on Large Corpora. In *Proceedings of International Conference in Computational Linguistics (COLING)*, pages 454–460, Nantes, France.